1)

Создание HTML5-формы

СОДЕРЖАНИЕ:

1. Элемент `<form>`
2. Группировка элементов формы `<fieldset>`
3. Создание полей формы `<input>`
4. Текстовые поля ввода `<textarea>`
5. Раскрывающийся список `<select>`
6. Надписи к полям формы `<label>`
7. Кнопки `<button>`
8. Флажки и переключатели в формах

1. Элемент <form>

Основу любой формы составляет элемент `<form>...</form>`. Он не предусматривает ввод данных, так как является контейнером, удерживая вместе все элементы управления формы - **поля.** Атрибуты этого элемента содержат информацию, общую для всех полей формы, поэтому в одну форму нужно включать поля, объединенные логически.

ТАБЛИЦА 1. АТРИБУТЫ ЭЛЕМЕНТА <FORM>

Атрибут	Значение / описание
`accept-charset`	Значение атрибута представляет собой разделенный пробелами **список кодировок символов**, которые будут использоваться для отправки формы, например, `<form accept-charset="ISO-8859-1">`.
`action`	**Обязательный атрибут**, который указывает `url` обработчика формы на сервере, которому передаются данные. Представляет из себя файл (например, `action.php`), в котором описано, что нужно делать с данными формы. Если значение атрибута не будет указано, то после перезагрузки страницы элементы формы примут значения по умолчанию. В случае, если вся работа будет выполняться на стороне клиента сценариями JavaScript, то для атрибута `action` можно указать значение `#`. Также можно сделать так, чтобы заполненная посетителем форма приходила вам на почту. Для этого нужно внести следующую запись: `<form action="mailto:адрес вашей электронной почты" enctype="text/plain"></form>`
`autocomplete`	Отвечает за запоминание введенных в текстовое поле значений и автоподстановку их при последующем вводе: `on` — означает, что поле не защищено, и его значение можно сохранять и извлекать, `off` — отключает автозаполнение для полей форм.
`enctype`	Используется для указания MIME-типа данных, отправляемых вместе с формой, например, `enctype="multipart/form-data"`. Указывается только в случае `method="post"`. `application/x-www-form-urlencoded` — тип содержимого по умолчанию, указывает на то, что передаваемые данные представляют список URL-кодированных переменных формы. Символы пробела (ASCII 32) будут закодированы как `+`, а специальный символ, например, такой как `!` будет закодирован шестнадцатиричной форме как `%21`. `multipart/form-data` — используется для отправки форм, содержащих файлы, не-ASCII данные и бинарные данные, состоит из нескольких частей, каждая из которых представляет содержимое отдельного элемента формы. `text/plain` — указывает на то, что передается обычный (не html) текст.
`method`	Задает способ передачи данных формы. Метод `get` передает данные на сервер через адресную строку браузера. При формировании запроса к серверу все переменные и их значения формируют последовательность вида `www.anysite.ru/form.php?var1=1&var2=2`. Имена и значения переменных присоединяются к адресу сервера после знака `?` и разделяются между собой знаком `&`. Все специальные символы и буквы, отличные от латинских, кодируются в формате `%nn`, пробел заменяется на `+`. Этот метод нужно использовать, если вы не передаете больших объемов информации. Если вместе с формой предполагается отправка какого-либо файла, этот метод не подойдет.

2. Группировка элементов формы

Элемент `<fieldset>...</fieldset>` предназначен для группировки элементов, связанных друг с другом, разделяя таким образом форму на логические фрагменты.

Каждой группе элементов можно присвоить название с помощью элемента `<legend>`, который идет сразу за открывающим тегом элемента `<fieldset>`. Название группы проявляется слева в верхней границе `<fieldset>`. Например, если в элементе `<fieldset>` хранится контактная информация:

```HTML
<form>
  <fieldset>
    <legend>Контактная информация</legend>
    <p><label for="name">Имя <em>*</em></label><input type="text" id="name"></p>
    <p><label for="email">E-mail</label><input type="email" id="email"></p>
  </fieldset>
<p><input type="submit" value="Отправить"></p>
</form>
```

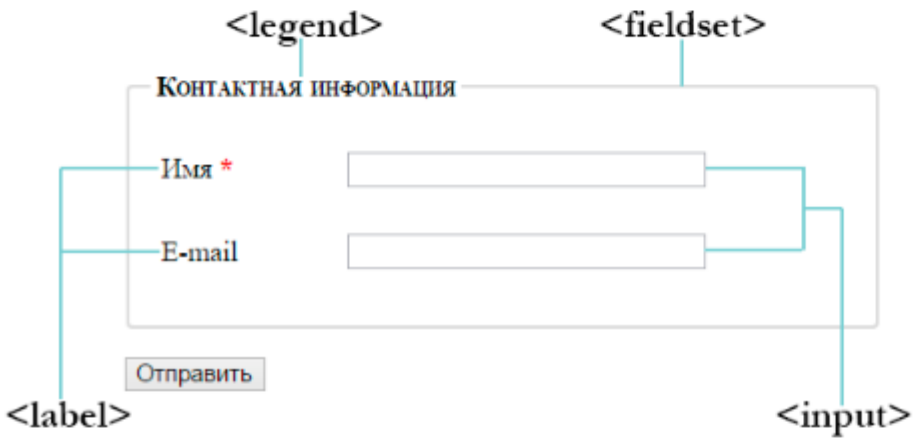

РИС. 2. ГРУППИРОВКА ЭЛЕМЕНТОВ ФОРМЫ С ПОМОЩЬЮ <FIELDSET>

ТАБЛИЦА 2. АТРИБУТЫ ЭЛЕМЕНТА <FIELDSET>

Атрибут	Значение / описание
`disabled`	Если атрибут присутствует, то группа связанных элементов формы, находящихся внутри контейнера `<fieldset>`, отключены для заполнения и редактирования. Используется для ограничения доступа к некоторым полям формы, содержащих ранее введенные данные. Атрибут используется без указания значения — `<fieldset disabled>`.
`form`	Значение атрибута должно быть равно атрибуту `id` элемента `<form>` в этом же документе. Указывает на одну или несколько форм, к которым принадлежит данная группа элементов. На данный момент атрибут не поддерживается ни одним браузером.
`name`	Определяет имя, которое будет использоваться для ссылки на элементы в JavaScript, или для ссылки на данные формы после заполнения и отправки формы. Является аналогом атрибута `id`.

3. Создание полей формы

Элемент `<input>` создает большинство полей формы. Атрибуты элемента отличаются в зависимости от типа поля, для создания которого используется этот элемент.

С помощью css-стилей можно изменить размер шрифта, тип шрифта, цвет и другие свойства текста, а также добавить границы, цвет фона и фоновое изображение. Ширина поля задается свойством `width`.

Атрибут	Значение / описание
`accept`	Определяет тип файла, разрешенных для отправки на сервер. Указывается только для `<input type="file">`. Возможные значения: file_extension — разрешает загрузку файлов с указанным расширением, например, `accept=".gif"`, `accept=".pdf"`, `accept=".doc"` `audio/*` — разрешает загрузку аудиофайлов `video/*` — разрешает загрузку видеофайлов `image/*` — разрешает загрузку изображений media_type — указывает на медиа-тип загружаемых файлов.
`alt`	Определяет **альтернативный текст** для изображений, указывается только для `<input type="image">`.
`autocomplete`	Отвечает за запоминание введенных в текстовое поле значений и автоподстановку их при последующем вводе: `on` — означает, что поле не защищено, и его значение можно сохранять и извлекать, `off` — отключает автозаполнение для полей форм.
`autofocus`	Позволяет сделать так, чтобы в загружаемой форме то или иное поле ввода уже имело фокус (было выбрано), являясь готовым к вводу значения.
`checked`	Атрибут проверяет, установлен ли флажок по умолчанию при загрузке страницы для полей типа `type="checkbox"` и `type="radio"`.
`disabled`	Отключает возможность редактирования и копирования содержимого поля.
`form`	Значение атрибута должно быть равно атрибуту `id` элемента `<form>` в этом же документе. Определяет форму, которой принадлежит данное поле.
`formaction`	Задает `url` файла, который будет обрабатывать введенные в поля данные при отправке формы. Задается только для полей типа `type="submit"` и `type="image"`. Атрибут переопределяет значение атрибута `action` самой формы.
`formenctype`	Определяет, как будут кодироваться данные полей формы при отправке на сервер. Переопределяет значение атрибута `enctype` формы. Задается только для полей типа `type="submit"` и `type="image"`. Варианты: `application/x-www-form-urlencoded` — значение по умолчанию. Все символы кодируются перед отправкой (пробелы заменяются на символ `+`, специальные символы преобразуются в значения ASCII HEX) `multipart/form-data` — символы не кодируются `text/plain` — пробелы заменяются на символ `+`, а специальные символы не кодируются.
`formmethod`	Атрибут определяет метод, который браузер будет использовать для отправки данных формы на сервер. Задается только для полей типа `type="submit"` и `type="image"`. Переопределяет значение атрибута `method` формы. Варианты: `get` — значение по умолчанию. Данные из формы (пара имя/значение) добавляются в url-адрес и отправляются на сервер: `URL?имя=значение&имя=значение` `post` — данные формы отправляются в виде http-запроса.
`formnovalidate`	Определяет, что данные полей формы не должны проверяться при отправке формы. Переопределяет значение атрибута `novalidate` формы. Можно использовать без указания значения атрибута.

`formtarget`	Определяет, где выводить ответ, полученный после отправки формы. Задается только для полей типа `type="submit"` и `type="image"`. Переопределяет значение атрибута `target` формы. `_blank` — загружает ответ в новое окно/вкладку `_self` — загружает ответ в то же окно (значение по умолчанию) `_parent` - загружает ответ в родительский фрейм `_top` - загружает ответ во весь экран framename - загружает ответ во фрейм с указанным именем.
`height`	Значение атрибута содержит количество пикселей без указания единицы измерения. Устанавливает высоту поля формы типа `type="image"`, например, `<input type="image" src="img_submit.gif" height="50">`. Рекомендуется одновременно устанавливать как высоту, так и ширину поля.
`list`	Является ссылкой на элемент `<datalist>`, содержит его `id`.Позволяет предоставить пользователю несколько вариантов на выбор, когда он начинает вводить значение в соответствующем поле.
`max`	Позволяет ограничить допустимый ввод числовых данных максимальным значением, значение атрибута может содержать целое или дробное число. Рекомендуется использовать этот атрибут вместе с атрибутом `min`. Работает со следующими типами полей: `number`, `range`, `date`, `datetime`, `datetime-local`, `month`, `time` и `week`.
`maxlength`	Атрибут задает максимальное количество символов, вводимых в поле. Значение по умолчанию 524288 символов.
`min`	Позволяет ограничить допустимый ввод числовых данных минимальным значением.
`multiple`	Позволяет пользователю ввести несколько значений атрибутов, разделяя их запятой. Применяется в отношении файлов и адресов электронной почты. Указывается без значения атрибута.
`name`	Определяет имя, которое будет использоваться для доступа к элементу `<form>`, к примеру, в таблицах стилей css. Является аналогом атрибута `id`.
`pattern`	Позволяет определять с помощью **регулярного выражения** синтаксис данных, ввод которых должен быть разрешен в определенном поле. Например, `pattern="[a-z]{3}-[0-9]{3}"` — квадратные скобки устанавливают диапазон допустимых символов, в данном случае — любые строчные буквы, число в фигурных скобках указывает, что нужны три строчные буквы, после которых следует тире, далее — три цифры в диапазоне от 0 до 9.
`placeholder`	Содержит текст, который отображается в поле ввода до заполнения (чаще всего это подсказка).
`readonly`	Не позволяет пользователю изменять значения элементов формы, выделение и копирование текста при этом доступно. Указывается без значения атрибута.
`required`	Выводит сообщение о том, что данное поле является обязательным для заполнения. Если пользователь попытается отправить форму, не введя в это поле требуемое значение, то на экране отобразится предупреждающее сообщение. Указывается без значения атрибута.
`size`	Задает видимую ширину поля в символах. Значение по умолчанию — 20. Работает со следующими типами полей: `text`, `search`, `tel`, `url`, `email` и `password`.

`type`	`button` — создает кнопку. Кнопка `checkbox` — превращает поле ввода во флажок, который можно установить или очистить, например, ☐ У меня есть автомобиль `color` — генерирует палитры цветов в поддерживающих браузерах, давая пользователям возможность выбирать значения цветов в шестнадцатеричном формате. `date` — позволяет вводить дату в формате дд.мм.гггг. День рождения: дд.мм.гггг `datetime-local` — позволяет вводить дату и время, разделенные прописной английской буквой `T` по шаблону дд.мм.гггг чч:мм. День рождения — день и время: дд.мм.гггг --:-- `email` — браузеры, поддерживающие данный атрибут, будут ожидать, что пользователь введет данные, соответствующие синтаксису адресов электронной почты. E-mail: `file` — позволяет загружать файлы с компьютера пользователя. Выберите файл: Выбор файла Не выбран ни один файл `hidden` — скрывает элемент управления, который не отображается браузером и не дает пользователю изменять значения по умолчанию. `image` — создает кнопку, позволяя вместо текста на кнопке вставить изображение. Перейти на сайт `month` — позволяет пользователю вводить год и номер месяца по шаблону гггг-мм. -------- ---- `number` — предназначено для ввода целочисленных значений. Его атрибуты `min`, `max` и `step` задают верхний, нижний пределы и шаг между значениями соответственно. Эти атрибуты предполагаются у всех элементов, имеющих численные показатели. Их значения по умолчанию зависят от типа элемента. Укажите количество (от 1 до 5): `password` — создает текстовые поля в форме, при этом вводимые пользователем символы заменяются на звездочки, маркеры, либо другие, установленные браузером значки. Введите пароль: `radio` — создает переключатель — элемент управления в виде небольшого кружка, который можно включить или выключить. Вегетарианец: ○ `range` — позволит создать такой элемент интерфейса, как ползунок, min / max — позволят установить диапазон выбора `reset` — создает кнопку, которая очищает поля формы от введенных пользователем данных. Сбросить

	`search` — обозначает поле поиска, по умолчанию поле ввода имеет прямоугольную форму. Поиск: [ ] `submit` — создает стандартную кнопку, активизируемую щелчком мыши. Кнопка собирает информацию с формы и отправляет ее для обработки. [Отправить] `text` — создает текстовые поля в форме, выводя однострочное текстовое поле для ввода текста. [ ] `time` — позволяет вводить время в 24-часовом формате по шаблону чч:мм. В поддерживающих браузерах оно отображается как элемент управления в виде числового поля ввода со значением, изменяемым с помощью мыши, и допускает ввод только значений времени. Укажите время: [--:--] `url` — поле предназначено для указания URL-адресов. Главная страница: [ ] `week` — соответствующий инструмент-указатель позволяет пользователю выбрать одну неделю в году, после чего обеспечит ввод данных в формате нн-гггг. В зависимости от года число недель может быть 52 или 53. Укажите неделю: [Неделя --, ----]
`value`	Определяет текст, отображаемый на кнопке, в поле или связанный текст. Не указывается для полей типа file.
`width`	Значение атрибута содержит количество пикселей. Позволяет задать ширину полей формы.

4. Текстовые поля ввода

Элемент `<textarea>...</textarea>` используется вместо элемента `<input type="text">`, когда нужно создать большие текстовые поля. Текст, отображаемый как исходное значение, помещается внутрь.

Размеры поля устанавливаются при помощи атрибутов `cols` - размеры по горизонтали, `rows` - размеры по вертикали. Высоту поля можно задать свойством `height`. Все размеры считаются исходя из размера одного символа моноширинного шрифта.

ТАБЛИЦА 4. АТРИБУТЫ ЭЛЕМЕНТА <TEXTAREA>

Атрибут	Значение / описание
`autofocus`	Устанавливает фокус на нужном начальном текстовом поле автоматически.
`cols`	Устанавливает ширину через количество символов. Если пользователь вводит больше текста, появляется полоса прокрутки.
`disabled`	Отключает возможность редактирования и копирования содержимого поля.
`form`	Значение атрибута должно быть равно значению атрибута `id` элемента `<form>` в этом же документе. Определяет одну или несколько форм, которым принадлежит данное текстовое поле.
`maxlength`	Значение атрибута задает максимальное число символов для ввода в поле.
`name`	Задает имя текстового поля.
`placeholder`	Определяет короткую текстовую подсказку, которая описывает ожидаемое вводимое значение.
`readonly`	Отключает возможность редактирования содержимого поля.
`required`	Выводит сообщение о том, что данное поле является обязательным для заполнения.
`rows`	Указывает число, которое означает, сколько строк должно отображаться в текстовой области.
`wrap`	Определяет, должен ли текст сохранять переносы строк при отправке формы. Значение `hard` сохраняет перенос, а значение `soft` не сохраняет. Если используется значение `hard`, то должно указываться значение атрибута `cols`.

5. Раскрывающийся список

Списки дают возможность расположить большое количество пунктов компактно. Раскрывающиеся списки создаются при помощи элемента `<select>...</select>`. Они позволяют выбрать одно или несколько значений из предложенного множества. По умолчанию в поле списка отображается его первый элемент.
Для добавления в список пунктов используются элементы `<option>...</option>`, которые располагаются внутри `<select>`.
Для систематизации списков применяется элемент `<optgroup>...</optgroup>`, который создает заголовки в списках.
Для списков возможно изменить размер шрифта, тип шрифта, цвет и другие свойства текста, а также добавить границы, цвет фона и фоновое изображение.

ТАБЛИЦА 5. АТРИБУТЫ ЭЛЕМЕНТА <SELECT>

Атрибут	Значение / описание
`autofocus`	Устанавливает автоматический фокус на элементе при загрузке страницы.
`disabled`	Отключает раскрывающийся список.
`form`	Определяет форму, которой принадлежит данный список. В качестве значения атрибута указывается идентификатор формы.
`multiple`	Дает возможность выбора одного или нескольких пунктов, для этого при выборе нужно нажать и удерживать нажатой клавишу `Ctrl`.
`name`	Определяет имя для выпадающего списка. Значение атрибута содержит название, отражающее тематику списка.
`required`	Выводит сообщение о том, что пользователь должен выбрать значение из раскрывающегося списка перед отправкой формы.
`size`	Задает количество одновременно видимых на экране элементов списка. Если количество элементов списка превышает установленное количество, появляется полоса прокрутки. Значение атрибута задается целым положительным числом.

ТАБЛИЦА 6. АТРИБУТЫ ЭЛЕМЕНТА <OPTION>

Атрибут	Значение / описание
`disabled`	Делает недоступным для выбора элемент списка.
`label`	Задает укороченную версию для элемента, которая будет отражаться в выпадающем списке. Значение атрибута содержит текст, описывающий соответствующий пункт выпадающего списка.
`selected`	Отображает выбранный элемент списка по умолчанию при загрузке страницы браузером.
`value`	Указывает значение, которое будет отправлено на сервер при отправке формы.

ТАБЛИЦА 7. АТРИБУТЫ ЭЛЕМЕНТА <OPTGROUP>

Атрибут	Значение / описание
`disabled`	Отключает данную группу элементов списка для выбора.
`label`	Задает заголовок для группы элементов выпадающего списка. Значение атрибута содержит текст, недоступный для выбора, который будет располагаться над соответствующими пунктами списка. Текст выделяется в браузере жирным начертанием.

6. Надписи к полям формы

Надписи к элементам формы создаются с помощью элемента `<label>...</label>`. Существует два способа группировки надписи и поля. Если поле находится внутри элемента `<label>`, то атрибут `for` указывать не нужно.

```
<!-- с указанием атрибута for -->
<label for="comments">Когда вы последний раз летали на самолете?</label>
<textarea id="comments"></textarea>

<!-- без атрибута for -->
<p><label>Кошка<input id="cat" type="checkbox"></label></p>
```

HTML

ТАБЛИЦА 8. АТРИБУТЫ ЭЛЕМЕНТА <LABEL>

Атрибут	Значение / описание
`for`	Определяет, к какому полю формы привязан данный элемент. Можно создавать поясняющие надписи к следующим элементам формы: `<input>`, `<textarea>`, `<select>`. Значение атрибута содержит идентификатор поля формы.

7. Кнопки

Элемент `<button>...</button>` создает кликабельные кнопки. В отличие от кнопок, созданных `<input>` (`<input type="submit"></input>`, `<input type="image">`, `<input type="reset">`, `<input type="button">`), внутрь элемента `<button>` можно поместить контент — текст или изображение. Для корректного отображения элемента `<button>` разными браузерами нужно указывать атрибут `type`, например, `<button type="submit"></button>`. Кнопки позволяют пользователям передавать данные в форму, очищать содержимое формы или предпринимать какие-либо другие действия. Можно создавать границы, изменять фон и выравнивать текст на кнопке.

ТАБЛИЦА 9. АТРИБУТЫ ЭЛЕМЕНТА <BUTTON>

Атрибут	Значение / описание
`autofocus`	Устанавливает фокус на кнопке при загрузке страницы.
`disabled`	Отключает кнопку, делая ее некликабельной.
`form`	Указывает на одну или несколько форм, которым принадлежит данная кнопка. Значение атрибута — идентификатор соответствующей формы.
`formaction`	Значение атрибута содержит url-адрес обработчика данных формы, отправляемых при нажатии на кнопку. Только для кнопки типа `type="submit"`. Переопределяет значение атрибута `action`, указанного для элемента `<form>`.
`formenctype`	Задает тип кодировки данных формы перед отправкой на сервер при нажатии на кнопки типа `type="submit"`. Переопределяет

	значение атрибута `enctype`, указанного для элемента `<form>`. Возможные значения: `application/x-www-form-urlencoded` — значение по умолчанию. Все символы перед отправкой будут закодированы. `multipart/form-data` — символы не кодируются. Используется в случае, когда с помощью формы загружаются файлы. `text/plain` — символы не кодируются, а пробелы заменяются на символ `+`.
`formmethod`	Атрибут определяет метод, который браузер будет использовать для отправки формы. Переопределяет значение атрибута `method`, указанного для элемента `<form>`. Указывается только для кнопок типа `type="submit"`. Возможные значения: `get` — данные из формы (пара имя/значение) добавляются в url-адрес и отправляются на сервер. Данный способ имеет ограничения по размеру отправляемых данных и не подходит для отправки паролей и конфиденциальной информации. `post` — данные из формы добавляются в виде http-запроса. Метод является более надежным и безопасным, чем `get` и не имеет ограничений по размеру.
`formnovalidate`	Атрибут задает, что данные формы не должны проверяться при отправке. Указывается только для кнопок типа `type="submit"`.
`formtarget`	Атрибут задает, в каком окне выводить результат после отправки формы. Указывается только для кнопок типа `type="submit"`. Переопределяет значение атрибута `target`, указанного для элемента `<form>`. `_blank` — загружает ответ в новое окно/вкладку `_self` — загружает ответ в то же окно (значение по умолчанию) `_parent` — загружает ответ в родительский фрейм `_top` — загружает ответ во весь экран `framename` — загружает ответ во фрейм с указанным именем.
`name`	Задает имя кнопки, значение атрибута — текст. Используется для ссылки на данные формы, после того как форма была отправлена, или для ссылки на данную кнопку (кнопки) в JavaScript.
`type`	Определяет тип кнопки. Возможные значения: `button` — кликабельная кнопка `reset` — кнопка сброса, возвращает первоначальное значение `submit` — кнопка для отправки данных формы.
`value`	Задает значение по умолчанию, отправляемое при нажатии на кнопку.

8. Флажки и переключатели в формах

Флажки в формах задаются с помощью конструкции `<input type="checkbox">`, а переключатель — при помощи `<input type="radio">`. Флажков, в отличие от переключателей, в одной форме может быть установлено несколько. Если для флажков указан атрибут `checked`, то при загрузке станицы на соответствующих полях формы флажки уже будут установлены.
Элемент `<label>` применяется при реализации выбора с помощью переключателей и флажков. Можно выбрать нужный пункт, просто щелкая кнопкой мыши на тексте, связанном с ним. Для этого нужно поместить `<input>` внутрь элемента `<label>`.

2) **Практическая работа**

Создайте форму для онлайн суши-кафе. Где сделайте несколько сгруппированных полей (fieldset)

1) Сведения о заказчике.
2) Выбор холодного ролла
3) Выбор горячего ролла
4) Выбор супа
5) Желаемое время доставки.
6) Инструкция курьеру (как проехать до вашего дома.